

# Systeembeheer met KDE

bron: [kde.org/areas/sysadmin](http://kde.org/areas/sysadmin)

vertaling: Tom Verbreyt

Herziening 0.1

Copyright © 2004 KDE-nl team

Dit document werd geschreven voor systeembeheerders die in hun organisatie gebruik maken van KDE.

## Inhoudsopgave:

### 1. De hiërarchie van het bestandssysteem

Dit hoofdstuk beschrijft hoe KDE omgaat met het bestandssysteem, waar het op zoek gaat naar bepaalde bestanden en waar het bepaalde bestanden bewaart. Het toont je ook hoe je die standaardinstellingen kan aanpassen.

1. Inleiding
2. Locatie van de boomstructuren
3. In de boomstructuur
4. Buiten de boomstructuur
5. Te doorzoeken locaties toevoegen

### 2. Configuratiebestanden

De configuratiebestanden van KDE zijn steeds eenvoudige tekstbestanden die uit groepjes sleutel/waarde-paren bestaan. In dit hoofdstuk gaat het onder meer over het integraal of gedeeltelijk beschermen van KDE's configuratiebestanden en over het gebruik van shell-variabelen.

1. Formaat & syntaxis
2. Trapsgewijze configuratie
3. Shell-substitutie
4. Localisatie
5. Afschermen van instellingen

### 3. Omgevingsvariabelen

Dit hoofdstuk beschrijft de kort de omgevingsvariabelen die je kan gebruiken om het gedrag van KDE en KDE-toepassingen te beïnvloeden.

### 4. Licenties en verdiensten

## 1. De hiërarchie van het bestandssysteem

[ [Inleiding](#) | [Locatie van de boomstructuren](#) | [In de boomstructuur](#) | [Buiten de boomstructuur](#) | [Te doorzoeken locaties toevoegen](#) ]

### 1.1 Inleiding

KDE bepaalt een hiërarchie binnen het bestandssysteem, die zowel door KDE zelf als door KDE-toepassingen gebruikt en gerespecteerd wordt. Algemeen gesproken gaat het om een eenduidige, vastgelegde mappenstructuur. Meestal zijn er daarvan twee: eentje op systeemniveau en eentje op gebruikersniveau, in de persoonlijke map van de gebruiker. Als systeembeheerder kan je echter bijkomende structuren aanmaken.

KDE en KDE-toepassingen zoeken bestanden door de verschillende mappenstructuren te doorzoeken. Die structuren [*nvdv*: ik noem ze vanaf nu gewoon *directory trees*, want echt natuurlijk klinkt ons fijne Nederlands niet altijd] zijn onderling gekenmerkt door een volgorde van prioriteit. Als een bestand aanwezig is in meerdere *directory trees*, dan krijgt het bestand in de eerstvermelde *tree* de voorkeur. Normaalgesproken is de *tree* die zich in de persoonlijke map van de gebruiker bevindt degene met de hoogste prioriteit. Alleszins is het ook daar dat wijzigingen opgeslagen zullen worden.

Ietwat anders gaat het met configuratiebestanden. Worden meerdere configuratiebestanden gevonden met dezelfde naam, dan wordt hun inhoud samengevoegd. Toch speelt ook hier de volgorde van prioriteit een rol: als twee bestanden dezelfde sleutel bepalen, dan zal het bestand met de hoogste voorkeur de waarde voor die sleutel instellen.

### 1.2 Locatie van de boomstructuren

De locatie van KDE's *directory trees* wordt bepaald door een aantal omgevingsvariabelen. In KDE 2.x waren er twee, sinds KDE 3.x zijn er daar nog twee bijgekomen.

### Omgevingsvariabelen in KDE 2.x:

- \$KDEHOME

Deze omgevingsvariabele bepaalt de plaats van de boomstructuur op gebruikersniveau en wordt door KDE (-toepassingen) gebruikt bij het aanmaken en bewaren van bestanden. De *tree* heeft de hoogste graad van prioriteit; bestanden of instellingen die op dit niveau aangetroffen worden, zullen vóór de bestanden of instellingen die in andere *trees* gevonden worden gelden.

Zoals de naam van de variabele al aangaf, zal deze *directory tree* zich normaalgesproken in de persoonlijke map van de gebruiker bevinden. Wordt de variabele niet van een nieuwe definitie voorzien, dan is de standaardlocatie \$HOME/.kde.

Als de variabele een waarde bevat die met een tilde begint (~), dan wordt die tilde *at runtime* vervangen door de persoonlijke map van de gebruiker. Let erop dat er correcte *quotes* gebruikt worden, anders zou de shell die substitutie al kunnen doorvoeren, wat ongewenst gedrag kan opleveren in combinatie met su.

- \$KDEDIR

Deze omgevingsvariabele bepaalt de plaats van de boomstructuur op systeemniveau. Alle gegevensbestanden (icoontjes, geluidsbestanden, bestanden die het menu vormgeven etcetera) die KDE-toepassingen vergezellen, worden normaalgezien in deze *directory tree* geïnstalleerd.

### Bijkomende omgevingsvariabelen in KDE 3.x:

- \$KDEROOTHOME

Vanaf KDE 3.x werd de omgevingsvariabele \$KDEROOTHOME geïntroduceerd, om problemen te voorkomen waarbij toepassingen die onder de root-gebruiker draaien ook bestanden met root-privileges opslaan in de persoonlijke map van de gebruiker. Toepassingen die onder uid 0 (root) draaien, zullen deze variabele dus gebruiken om te weten te komen waar de gebruikersmap zich bevindt en waar ze hun bestanden mogen opslaan. Wordt deze variabele niet vastgelegd, dan wordt de persoonlijke map van de root-gebruiker opgezocht in het password-bestand. Daaraan wordt ".kde" toegevoegd, zodat het standaard om /root/.kde/ gaat.

- \$KDEDIRS

Sinds KDE 3.x is het mogelijk om meer dan één *directory tree* op systeemniveau op te geven. Zo kan je groepen van gebruikers een specifieke *directory tree* toewijzen, die bijvoorbeeld bijkomende toepassingen bevat, speciale bronnen voor bepaalde toepassingen of een specifiek geheel van standaardinstellingen die voor die gebruikersgroep het best geschikt is. Het is beter om standaardinstellingen op deze manier vast te leggen (en niet via een /etc/skel-kluwen), omdat je zo ook wijzigingen aan de standaardconfiguratie kan toebrengen nadat gebruikers aangemaakt zijn.

De mappen die je in \$KDEDIRS opgeeft, scheidt je met een dubbele punt (:). De *directories* worden opgesomd in volgorde van prioriteit: de eerste geldt als de belangrijkste, de laatste heeft de laagste voorkeur. Omdat een *directory tree* voor een groep van gebruikers normaalgezien de instellingen op systeemniveau zou moeten overschrijven, wordt de *directory tree* op groepsniveau dus best vóór de *tree* op systeemniveau vermeld.

Blijft de variabele \$KDEDIRS onbepaald, dan wordt de locatie van \$KDEDIR gebruikt.

Om toepassingen van KDE 2.x in een KDE 3.x-omgeving te gebruiken, moet je \$KDEDIR naar de plaats van de KDE 2.x-installatie en \$KDEDIRS naar de plaats van de KDE 3.x-installatie laten verwijzen.

In de dagelijkse communicatie wordt met \$KDEHOME naar de *directory tree* op gebruikersniveau verwezen en met \$KDEDIR naar de *tree* op systeemniveau.

### Voorbeeld

Een leidinggevend figuur van een universiteit zou deze instellingen kunnen hebben:

```
KDEHOME='~/ .kde3'
KDEROOTHOME='/root/.kde3'
KDEDIRS='/opt/kde_staff:/opt/kde3'
```

In dat voorbeeld worden gebruikersinstellingen opgeslagen in de map `.kde3` in de persoonlijke map van de gebruiker. Toepassingen die als root draaien, bewaren hun instellingen in `/root/.kde3`. KDE 3 is geïnstalleerd in `/opt/kde3`, maar er is een bijkomende *directory tree* onder `/opt/kde_staff`. Configuratiebestanden die zich daar bevinden, zullen een hogere prioriteit bezitten dan de bestanden in de systeemmappen onder `/opt/kde3`. `/opt/kde_staff` zou bijkomende toepassingen kunnen bevatten, die enkel voor belangrijke personen beschikbaar zouden mogen zijn.

### 1.3 In de boomstructuur

Elke *directory tree* waarvan KDE gebruik maakt, is gekenmerkt door een vaste structuur. Mappen die niet relevant zijn voor een bepaalde *tree* kunnen echter weggelaten worden: de mappen die gebruikt worden voor het bewaren van tijdelijke bestanden vind je bijvoorbeeld meestal onder `$_KDEHOME` terug, maar niet in de andere *trees*.

KDE voegt de submappen in de verschillende boomstructuren samen en gebruikt het geheel als één enkele KDE-bron. De namen van KDE-bronnen (*KDE resources*) worden hieronder in tabellen beschreven.

Het gaat grofweg om drie grote categorieën: bestanden die specifiek gelden voor de CPU/architectuur, bestanden die gebonden zijn aan de betreffende host en bestanden die noch met de host, noch met de CPU, noch met de architectuur onlosmakelijk verbonden zijn.

#### CPU/architectuur-specifieke mappen:

Map	KDE-bron	Beschrijving
bin/	exe	Voor uitvoerbare bestanden van KDE.
cgi-bin/	cgi	CGI-scripts die door het KDE Help Center gebruikt kunnen worden.
lib/	lib	Voor KDE-bibliotheken.
lib/kde3/	module	Voor componenten, plugins en andere objecten die <i>at runtime</i> gebruikt kunnen worden door KDE 3.x-toepassingen.

#### Host-specifieke mappen:

Deze mappen zijn enkel beschikbaar onder `$_KDEHOME` en zijn normaalgezien gesymmlinked naar plekken buiten de `$_KDEHOME-tree`.

Map	KDE-bron	Beschrijving
socket-<HOSTNAME>	socket	Deze map bevat communicatie-sockets. Het bestandssysteem waarvan <code>\$_KDEHOME</code> gebruik maakt kan daartoe ongeschikt zijn, daarom verwijst deze map standaard naar een andere locatie.
tmp-<HOSTNAME>	tmp	Deze map bevat tijdelijke bestanden. Het bestandssysteem waarvan <code>\$_KDEHOME</code> gebruik maakt, zou zich op het netwerk kunnen bevinden. Om systeemprestaties niet onnodig negatief te beïnvloeden, verwijst deze map standaard naar een andere, lokaal gelegen plek.
cache-<HOSTNAME>	cache	Nieuw in KDE 3.2. Deze map bevat gecachete informatie, zoals HTTP-objecten, geformateerde help-pagina's en de cache van de systeemconfiguratie ( <code>ksycoca</code> ). Deze informatie is niet strikt noodzakelijk, daarom verwijst deze map standaard naar een andere locatie buiten <code>\$_KDEHOME</code> , om het maken van

reservekopieën niet nodeloos te bemoeilijken en om schijfruimte te besparen.

### En verder...

De meerderheid van de systeemmappen bevat gegevens die niet specifiek CPU/architectuur- of hostgebonden zijn. Al deze mappen worden voorafgegaan door /share.

Map	KDE-bron	Beschrijving
share/applnk/	apps	Bevat .desktop-bestanden die het KDE-menu beschrijven.
share/apps/	data	Bevat toepassings specifieke gegevens. Elke toepassing heeft hier een eigen submap waarin hij zijn bestanden kan bewaren.
share/config/	config	Bevat configuratiebestanden. Configuratiebestanden worden meestal genoemd naar de toepassing waartoe ze behoren, gevolgd door "rc". Er zijn ook component-specifieke bestanden, waarheen dus verwezen wordt door alle toepassingen die van die component gebruik maken. Een geval apart is <code>kdeglobals</code> , dat door alle KDE-toepassingen ingelezen wordt.
share/config/session/	-	Deze map wordt gebruikt bij het sessiebeheer en is normaalgezien enkel voorhanden onder <code>\$KDEHOME</code> . Aan het einde van een sessie bewaren KDE-toepassingen hier hun status. De bestandsnamen beginnen met de toepassingsnaam, gevolgd door een getal. De sessie-beheerder, <code>ksmserver</code> , slaat verwijzingen naar deze getallen op bij het bewaren van sessies in <code>ksmserverrc</code> .
share/doc/HTML/	html	Bevat de documentatie van KDE-toepassingen. Documentatie wordt gecategoriseerd volgens taal en naar de toepassingen waartoe hij behoort. Normalgezien bevat deze map minstens twee bestanden: "index.docbook", dat de documentatie in ongeformateerd <i>DocBook</i> -formaat bevat, en "index.cache.bz2", dezelfde documentatie in de vorm van met <code>bzip2</code> gecomprimeerd HTML. De HTML-versie wordt gebruikt door <code>khelpcenter</code> . Is de HTML-versie niet beschikbaar, dan wordt die opnieuw gegenereerd op basis van de <i>DocBook</i> -versie, maar dat vergt wat tijd.
share/icons/	icon	Bevat iconen, gecategoriseerd volgens uiterlijke stijl, afmetingen en gebruikscategorie.
share/mimelnk	mime	Bevat .desktop-bestanden die mimetypes beschrijven.
share/services/	services	Bevat .desktop-bestanden die "services" beschrijven. "Services" en toepassingen lijken erg op elkaar; het belangrijkste verschil is dat services normaalgezien enkel gebruikt worden door toepassingen of door andere services, terwijl toepassingen meestal door de gebruiker eigenhandig opgestart worden. Services worden niet opgesomd in het KDE-menu.
share/servicetypes/	servicetypes	Bevat .desktop-bestanden die "servicetypes" beschrijven. Een "servicetype" staat meestal voor een bepaalde programmeer-interface. Toepassingen en services beschrijven in hun .desktop-bestanden welk servicetype ze aanbieden.
share/sounds/	sound	Bevat geluidsbestanden.

share/templates/	templates	Bevat sjablonen voor verschillende bestandstypes. Een sjabloon ( <i>template</i> ) bestaat uit een <code>.desktop</code> -bestand dat het bestandstype beschrijft en een verwijzing bevat naar een ander bestand in de submap <code>.source</code> . De sjablonen in deze map verschijnen in het menu onder "Create New..." ("Nieuw..."), beschikbaar op het bureaublad en in de bestandsbrowser Konqueror. Wanneer de gebruiker een sjabloon kiest, wordt het bronbestand gekopieerd.
share/wallpapers/	wallpaper	Bevat afbeeldingen die gebruikt kunnen worden als bureaublad-achtergrond.

#### 1.4 Buiten de boomstructuur

Zoals hogerop reeds gezegd werd, zijn er drie host-specifieke mappen die meestal middels een symlink naar andere locaties verwijzen. Als die locaties niet bestaan, zullen onderstaande symlinks en mappen aangemaakt worden met `lnusertemp`. Omdat zowel `/tmp` als `/var/tmp` *world writable* (door eender wie beschrijfbaar) zijn, bestaat de kans dat de mappen al bestaan, maar een andere gebruiker toebehoren. In dat geval zal `lnusertemp` een nieuwe map met een alternatieve naam aanmaken, en dan natuurlijk daarheen laten verwijzen.

Symlink	Verwijst naar...	Beschrijving	Aangemaakt met...
<code>\$KDEHOME/socket-&lt;HOSTNAME&gt;</code>	<code>/tmp/ksocket-&lt;GEBRUIKER&gt;/</code>	<p>Maakt een map aan voor lokale communicatie-sockets en een symlink die daarheen verwijst. De samengevoegde lengte van de naam van de map en eender welke communicatie-socket mag niet langer zijn dan 106 tekens.</p> <p>Standaard wordt deze map aangemaakt onder <code>/tmp</code>; je kan andere locaties instellen door middel van de omgevingsvariabele <code>\$KDETMP</code>.</p>	<code>lnusertemp socket</code>
		<p>Maakt een map aan voor tijdelijke bestanden en een symlink die daarheen verwijst. Ter wille van de prestatie van het systeem is het aan te bevelen deze map op een lokaal bestandssysteem te hebben, maar dat is</p>	

<p>\$KDEHOME/tmp - &lt;HOSTNAME&gt;</p>	<p>/tmp/kde - &lt;GEBRUIKER&gt;/</p>	<p>niet strikt noodzakelijk.</p> <p>In KDE versies tot en met KDE 3.1 bevindt zich hier ook de cache van de systeemconfiguratie (ksycoca en ksycocastamp). Het is aan te bevelen deze bestanden NIET te verwijderen bij het opstarten van het systeem, omdat dat het opstarten van KDE erg zal vertragen.</p> <p>Standaard wordt deze map aangemaakt onder /tmp; je kan andere locaties instellen door middel van de omgevingsvariabele \$KDETMP.</p>	<p>lnusertemp tmp</p>
<p>\$KDEHOME/cache - &lt;HOSTNAME&gt;</p>	<p>/var/tmp/kdecache - &lt;GEBRUIKER&gt;/</p>	<p>Nieuw in KDE 3.2. Maakt een map aan voor cache-bestanden en een symlink die daarheen verwijst. Ter willen van de prestatie van het systeem is het aan te bevelen deze map op een lokaal bestandssysteem te hebben, maar dat is niet strikt noodzakelijk.</p> <p>Sinds KDE 3.2 bevindt zich hier ook de cache van de systeemconfiguratie (ksycoca en ksycocastamp).</p> <p>Standaard wordt deze map aangemaakt onder /var/tmp; je kan andere locaties instellen door middel van de omgevingsvariabele \$KDEVARTMP.</p>	<p>lnusertemp cache</p>

deel uitmaken van KDE's *desktop communication protocol* DCOP. Het gaat om volgende bestanden:

- `$HOME/.ICEauthority`  
Dit bestand bevat *access tokens* voor DCOP. Toepassingen moeten geschikte *access tokens* kunnen voorleggen om via DCOP en de *dcopserver* met andere toepassingen te kunnen communiceren. Wanneer de *dcopserver* gestart wordt, zal hij een nieuw *access token* genereren en opslaan in dit bestand. Met de omgevingsvariabele `$ICEAUTHORITY` kan je een ander bestand voor dit doel bestemmen.
- `$HOME/.DCOPserver_${HOSTNAME}_${DISPLAY}`  
Dit bestand bepaalt hoe de *dcopserver* van een bepaalde host en voor een bepaalde X display benaderd moet worden. Sinds KDE 3.1.4 kan je met de omgevingsvariabele `$DCOPAUTHORITY` kan je een ander bestand voor dit doel bestemmen. Merk echter op dat elke gelijktijdige KDE-sessie een eigen, uniek bestand moet hebben.

In de context van een **Kiosk**-omgeving kan het nuttig zijn om persoonlijke mappen *read-only* (enkel-lezen) te maken. In dat geval moeten `$ICEAUTHORITY` en `$DCOPAUTHORITY` naar beschrijfbare locaties verwijzen, want KDE zal weigeren op te starten als de persoonlijke map niet beschrijfbaar is. Sinds KDE 3.1.4 kan je dat gedrag vermijden door de omgevingsvariabele `$KDE_HOME_READONLY` in te stellen op eender welke niet-lege waarde.

Er zijn meldingen bekend die erop wijzen dat persoonlijke mappen die gemount zijn op bepaalde versies van Netware-bestandssystemen geen hardlinks ondersteunen. Dat veroorzaakt problemen bij het genereren van `$HOME/.ICEauthority` en leidt dus tot de onmogelijkheid om in te loggen in KDE. Je kan de situatie oplossen door `$ICEAUTHORITY` naar een bestand buiten de persoonlijke map te laten verwijzen, bijvoorbeeld ergens onder `/tmp`. Merk op dat elke gebruiker zijn eigen unieke bestand moet hebben.

### 1.5 Te doorzoeken locaties toevoegen

KDE-toepassingen zoeken gegevens op door middel van de *resource names* (bronnen-namen) die reeds beschreven werden in de sectie [In de boomstructuur](#). KDE zal deze namen omzetten naar eigenlijke mappen, door de locaties van mappenstructuren te combineren met de mappen die opgesomd werden in de hogerop opgestelde tabellen.

#### Voorbeeld

Een gebruiker heeft volgende instellingen van *directory trees*:

```
KDEHOME='~/ .kde3'
KDEDIRS='/opt/kde_staff:/opt/kde3'
```

Zoekt een toepassing nu naar een "wallpaper"-bestand, dan wordt aan elk van de vermelde *directory trees* `share/wallpapers/` toegevoegd. Daarna worden alle resulterende mappen op het bestand doorzocht:

- `~/ .kde3/share/wallpapers/`
- `/opt/kde_staff/share/wallpapers/`
- `/opt/kde3/share/wallpapers/`

Door meer *directory trees* aan de omgevingsvariabele `$KDEDIRS` toe te voegen, kan je meerdere mappen laten doorzoeken. Vaak is het wenselijk om maar één enkele map voor een zoektocht toe te voegen en geen hele mappenstructuur. Afzonderlijke mappen kan je toevoegen in het configuratiebestand `kdeglobals`, in de sectie "Directories". Wijs daartoe één of meerdere mappen toe aan de sleutel "dir\_", gevolgd door de naam van de KDE-bron. Meerdere mappen scheid je met een komma (,).

#### Voorbeeld

Om de map `/data/photos` toe te voegen aan de systeembron "wallpapers", zet je deze twee lijnen in `kdeglobals`:

```
[Directories]
dir_wallpaper=/data/photos
```

Wanneer een toepassing nu op zoek gaat naar "wallpaper"-bestanden, dan zal hij deze locaties raadplegen:

- /data/photos
- ~/.kde3/share/wallpapers/
- /opt/kde\_staff/share/wallpapers/
- /opt/kde3/share/wallpapers/

## 2. Configuratiebestanden

[ [Formaat & syntaxis](#) | [Trapsgewijze configuratie](#) | [Shell-substitutie](#) | [Localisatie](#) | [Afschermen van instellingen](#) ]

### 2.1 Formaat & syntaxis

Voor al zijn configuratiebestanden maakt KDE gebruik van eenvoudige, "platte" tekstbestanden, die uit gegroepeerde sleutel/waarde-paren (*key-value pairs*) bestaan. Alle configuratiebestanden gebruiken UTF-8 als er behoefte is aan lettertekens buiten de ASCII-set.

Het begin van een groep wordt aangegeven door de naam van die groep, omsloten door vierkante haken, gevolgd door de sleutel/waarde-paren die tot die groep behoren. De groep wordt afgesloten als een nieuwe groep begint of als het einde van het bestand bereikt wordt.

Instellingen die aan het begin van het bestand staan en niet door een groepsnaam voorafgegaan worden, maken deel uit van de standaard-groep.

#### Voorbeeld

Het volgende voorbeeld is een configuratiebestand dat uit twee groepen bestaat. De eerste groep bevat de sleutels `LargeCursor` en `SingleClick`, de tweede groep bevat de sleutels `Show hidden files` en `Sort by`.

```
[KDE]
LargeCursor=false
SingleClick=true

[KFileDialog Settings]
Show hidden files=false
Sort by=Name
```

Items binnen een groep bestaan uit een sleutel en een waarde, gescheiden door een is-gelijk-aan-teken. De sleutel mag spaties bevatten en mag gevolgd worden door opties binnen vierkante haken. Het deel na het gelijkheidsteken is de waarde van het item. Witruimte die voor of na het gelijkheidsteken staat, wordt genegeerd, net als eventuele witruimte die zich achter het item bevindt.

Wordt een waarde wel degelijk verondersteld een spatie aan het begin of aan het einde te hebben, dan kan die weergegeven worden met een backslash (\), gevolgd door een "s". Er zijn ook andere backslash-codes mogelijk:

- \s kan gebruikt worden als een spatie
- \t kan gebruikt worden als een tab
- \r kan gebruikt worden als *carriage return*
- \n kan gebruikt worden als *linefeed*-karakter (nieuwe lijn)
- \\ kan gebruikt worden om de backslash zelf weer te geven

#### Voorbeeld

Het volgende voorbeeld levert hetzelfde resultaat op als dat van het vorige voorbeeld. Witruimte die de sleutels en/of waarden omringt, wordt genegeerd.

```
[KDE]
LargeCursor=      false
SingleClick = true

[KFileDialog Settings]
  Show hidden files  = false
```

Sort by =Name

## Voorbeeld

In onderstaand voorbeeld begint de waarde van het item `Caption` met twee spaties, het item `Description` bestaat uit drie regels: *linefeeds* met backslash-notatie scheiden de verschillende regels.

```
[Preview Image]
Caption=\s My Caption
Description=This is\na very long\ndescription.
```

Lege lijnen worden genegeerd, net als lijnen die voorafgegaan worden door een *hash mark* (`#`). Je kan het hash-teken gebruiken om commentaar toe te voegen; merk echter op dat commentaar niet bewaard blijft als een KDE-toepassing zijn configuratiebestanden bijwerkt!

## 2.2 Trapsgewijze configuratie

Er kunnen meerdere configuratiebestanden met dezelfde naam voorkomen in de `share/config/`-submappen van de verschillende KDE-mappenstructuren. In dat geval wordt de informatie uit al die configuratiebestanden samengevoegd, sleutel per sleutel. Wordt een bepaalde sleutel binnen een bepaalde groep op meerdere plekken gedefinieerd, dan maakt KDE gebruik van de waarde die in de *directory tree* met de hoogste prioriteit aangetroffen werd. Configuratiebestanden in `$HOME` genieten altijd de voorkeur.

Wordt een bepaalde sleutel binnen een bepaalde groep meerdere keren gedefinieerd binnen één enkel bestand, dan wordt de waarde van het laatste voorkomen van die sleutel gebruikt.

Dankzij dit trapsgewijs afdalende configuratiesysteem is het mogelijk om gebruikers bepaalde standaardinstellingen aan te bieden (op systeemniveau of binnen de hele organisatie), zonder hen de mogelijkheid te ontnemen om individuele wijzigingen aan die instellingen toe te brengen. KDE zal de instellingen die op een hoger niveau bepaald worden nooit wegschrijven in de configuratiebestanden van gebruikers, onder `$KDEHOME`, zodat systeemwijzigingen meteen van kracht zijn voor alle gebruikers. Merk op dat die aanpak verschilt van een constructie met `/etc/skel`; wijzigingen die je daar toebrenget, zullen enkel beginnen gelden voor nieuw aangemaakte gebruikersaccounts.

Soms is het niet de bedoeling dat afzonderlijke gebruikers de standaardinstellingen kunnen aanpassen. In zo'n geval kan je de standaardinstelling vastleggen.

## 2.3 Shell-substitutie

Met "shell-substitutie" (*shell expansion*) is het mogelijk om standaardwaarden een dynamischer karakter te geven: de waarde van een configuratie-sleutel kan zo ingesteld worden, dat hij afhangt van een omgevingsvariabele of van de uitvoer van een shell-commando.

Om shell-substitutie mogelijk te maken voor een configuratie-item, zet je `[$e]` achter de sleutel van dat item. Normaalgezien wordt de gesubstitueerde vorm na het eerste gebruik opgeslagen in het configuratiebestand van de gebruiker. Om dat te vermijden, kan je de instelling vastleggen door `[$ie]` te gebruiken.

## Voorbeeld

In dit voorbeeld wordt de waarde van het item `Host` bepaald door de uitvoer van het programma `hostname`. De instelling wordt beveiligd, zodat de waarde zeker telkens opnieuw dynamisch bepaald wordt.

De waarde van het item `Email` wordt ingevuld met de waarden van de omgevingsvariabelen `$USER` en `$HOST`. Als gebruiker "pino" ingelogd is op host "sesamstraat", dan zal dat de waarde "pino@sesamstraat" opleveren. Deze waarde wordt niet vastgelegd.

```
[Mail Settings]
Host[$ie]=$(hostname)
Email[$e]=${USER}@{HOST}
```

## 2.4 Localisatie

Je kan alle configuratie-items indexeren met een taalcode. De waarde die de gebruiker aangegeven heeft, wordt gebruikt om de corresponderende sleutelwaarde op te zoeken. Als de standaard-taalinstelling geselecteerd is (Amerikaans Engels) of als er geen index voorhanden is die met de gekozen taal overeenstemt, dan wordt de sleutelwaarde zonder index gebruikt.

### Voorbeeld

In dit voorbeeld hangt de waarde van het item `Caption` af van de gekozen taal. Als de gebruiker voor Frans gekozen heeft (taalcode "fr"), dan wordt de waarde van het item "Ma légende". In alle andere gevallen wordt de waarde "My Caption" gebruikt.

```
[Preview Image]
Caption=My Caption
Caption[fr]=Ma Légende
```

## 2.5 Afschermen van instellingen

Om te vermijden dat gebruikers standaardinstellingen overschrijven, kan je die instellingen beveiligen. Je kan instellingen individueel, per groep of per bestand afschermen. Een afzonderlijk item leg je vast door `[$i]` toe te voegen achter de sleutel. Een groep configuratie-items leg je vast door `[$i]` toe te voegen achter de naam van de groep. Om een heel bestand af te schermen, zet je `[$i]` op de verder lege eerste lijn van het bestand.

### Voorbeeld

Hieronder is het item `Caption` vastgelegd, net als alle instellingen binnen de groep `Mail Settings`.

```
[Preview Image]
Caption[$i]=My Caption
```

```
[Mail Settings][$i]
Host[$e]=$hostname
Email=pino@sesamstraat.biz
```

Als je op die manier een instelling hebt beveiligd, dan worden de items die normaalgezien een hogere prioriteit zouden krijgen gewoon genegeerd.

Ook de wijzigingen die sommige toepassingen aan vastgelegde instellingen zouden willen toebrengen, worden zonder enige waarschuwing genegeerd. Het hangt van de toepassing af hoe op beveiligde instellingen gereageerd wordt. Sommige toepassingen herkennen een dergelijke situatie: ze merken dat de instelling in kwestie centraal afgeschermd wordt, en verwijderen daarom de elementen die normaalgezien controle over de vastgelegde instellingen zouden bieden uit hun gebruikersinterface. Andere toepassingen zullen die instellingen blijven aanbieden, ofschoon ze helemaal geen impact meer hebben. Soms is het echter nog steeds mogelijk om de instelling aan te passen in het draaiende programma. De wijziging blijft dan gelden zolang het programma blijft draaien. Zodra het opnieuw gestart wordt, zal de beveiligde instelling terug van kracht zijn.

## 3. Omgevingsvariabelen

Wat volgt is een opsomming van de omgevingsvariabelen die KDE respecteert of verwacht.

### KDEDIR

Moet gedefinieerd worden als `KDEDIRS` niet bepaald is en moet naar de basis van de KDE-installatie wijzen. KDE vindt er zijn gegevens, zoals bibliotheken, iconen etc.

### KDEDIRS

Overschrijft `KDEDIR` en maakt het je mogelijk om meerdere mappen op te geven waarin KDE zijn gegevens moet gaan zoeken. Nuttig als je programma's wil of moet installeren op een andere plek dan de rest van je KDE-installatie.

### KDEHOME

Als `KDEHOME` niet bepaald wordt, gebruikt KDE `~/ .kde` om individuele gegevens van gebruikers op te slaan.

### KDEROOTHOME

Als `KDEROOTHOME` niet bepaald wordt, gebruikt KDE `/root/.kde` om individuele gegevens van de root-gebruiker op te slaan. Werd geïntroduceerd om te voorkomen dat KDE abusievelijk gebruikersgegevens met rootrechten overschreef wanneer de gebruiker een KDE-programma draaide nadat hij of zij met `su root` geworden was.

#### **KDEWM**

Als `KDEWM` bepaald wordt, wordt het als window manager voor KDE gebruikt in het script `startkde`, in plaats van `kwin`.

#### **KDE\_LANG**

Overschrijft de taalinstellingen van KDE, dus `KDE_LANG=fr kprogram` & zou het programma `kprogram` in het Frans opstarten als de daarvoor benodigde bestanden geïnstalleerd zijn, ongeacht de instellingen die bepaald zijn via het configuratiecentrum.

#### **KDE\_DISPLAY**

Een verouderde (?) manier om `DISPLAY` in te stellen voor meerdere beeldschermen.

#### **KDE\_MULTITHREAD**

Stel deze variabele in op "true" om aan te geven dat KDE op een systeem met meerdere beeldschermen draait.

#### **KDE\_DEBUG**

Stel deze variabele in om de KDE Crash Handler uit te schakelen (hetzelfde effect krijg je met de commandoregel-optie `--nocrashhandler`).

#### **KDE\_FORK\_SLAVES (vanaf KDE 3.2.3)**

Stel deze variabele in om KIO-slaves direct vanuit het eigen toepassingsproces te laten starten. Standaard worden KIO-slaves via `klauncher/kdeinit` opgestart. Deze optie kan nuttig zijn als de KIO-slave in dezelfde omgeving als de toepassing moet draaien, zoals in het geval van Clearcase.

#### **KDE\_HOME\_READONLY**

Stel deze variabele in om aan te geven dat de persoonlijke mappen alleen-lezen gekoppeld zijn (*mounted as read-only*).

#### **KDE\_NO\_IPV6 (vanaf KDE 3.2.3)**

Stel deze variabele in om IPv6-ondersteuning en IPv6 DNS-lookups uit te schakelen.

#### **KDE\_IS\_PRELINKED**

Stel deze variabele in om aan te geven dat je KDE-binaries en -bibliotheken *prelinked* zijn.

#### **KDE\_MALLOC**

Als deze variabele op "1" ingesteld staat, dan wordt de snelle `malloc`-routine uit `kdecore` gebruikt als KDE gecompileerd is met de optie `--enable-fast-malloc`. Anders wordt de `libc`-routine gebruikt.

#### **KDE\_NOUNLOAD**

Als deze variabele ingesteld is, wordt `KLibLoader` opgedragen om dynamisch geladen bibliotheken nooit te verwijderen.

#### **KDE\_DOWNLOAD**

Als deze variabele ingesteld is, wordt `KLibLoader` opgedragen om dynamisch geladen bibliotheken altijd te proberen te verwijderen. Opgelet, dat leidt waarschijnlijk tot crashes!

#### **KDE\_UTF8\_FILENAMES**

Als deze variabele ingesteld is, gaat KDE ervan uit dat alle bestandsnamen in UTF-8 gecodeerd zijn, ongeacht de huidige C-locale.

#### **KDE\_FULL\_SESSION (vanaf KDE 3.2)**

Deze variabele wordt automatisch op "true" ingesteld bij het opstarten van KDE. Ze wordt onder andere door Konqueror gebruikt om te weten of het moet overwegen in het geheugen geladen te blijven voor toekomstig hergebruik, als het gesloten wordt. Is de variabele niet ingesteld, dan zal Konqueror helemaal verdwijnen als het afgesloten wordt.

#### **KDESYCOCA**

Biedt de mogelijkheid om pad en bestandsnaam van het gegeneerde KDE-systeemconfiguratie-cachebestand (is dat een correct Nederlandstalig woord?) op te geven.

#### **KDETMP**

Staat je toe om een ander pad dan `/tmp` op te geven om er KDE zijn tijdelijke bestanden te laten opslaan.

#### **KDEVARTMP**

Staat je toe om een ander pad dan `/var/tmp` op te geven om er KDE zijn variabele bestanden te laten opslaan.

#### **XDG\_DATA\_HOME (vanaf KDE 3.2)**

Bepaalt de basismap, ten opzichte waarvan gebruikersspecifieke gegevensbestanden opgeslagen moeten worden. Als `$XDG_DATA_HOME` leeg of onbepaald is, wordt standaard `$HOME/.local/share` gebruikt (cf. de specs).

#### **XDG\_CONFIG\_HOME (vanaf KDE 3.2)**

Bepaalt de basismap, ten opzichte waarvan gebruikersspecifieke configuratiebestanden opgeslagen moeten worden. Als \$XDG\_CONFIG\_HOME leeg of onbepaald is, wordt standaard \$HOME/.config gebruikt (cf. [de specs](#)).

**XDG\_DATA\_DIRS (vanaf KDE 3.2)**

Bepaalt de op voorkeur gesorteerde reeks basismappen die op zoek naar gegevensbestanden doorzocht moeten worden, samen met de basismap van \$XDG\_DATA\_HOME. De opgegeven mappen moeten gescheiden worden met een dubbele punt (":"). Als \$XDG\_DATA\_DIRS leeg of onbepaald is, wordt standaard /usr/local/share/:/usr/share/ gebruikt (cf. [de specs](#)).

**XDG\_CONFIG\_DIRS (vanaf KDE 3.2)**

Bepaalt de op voorkeur gesorteerde reeks basismappen die op zoek naar configuratiebestanden doorzocht moeten worden, samen met de basismap van \$XDG\_CONFIG\_HOME. De opgegeven mappen moeten gescheiden worden met een dubbele punt (":"). Als \$XDG\_CONFIG\_DIRS leeg of onbepaald is, wordt standaard /etc/xdg gebruikt. (cf. [de specs](#)).

#### 4. Licenties en verdiensten

De originele documenten waarvan u de vertaling net achter de kiezen heeft, vindt u terug op <http://www.kde.org/areas/sysadmin/>, samen met nog enkele andere teksten die zich in de context van systeembeheer met KDE situeren.

Het is toegestaan dit document te kopiëren, te verdelen en/of te wijzigen onder de voorwaarden van de GNU vrije documentatielicentie, versie 1.1 of een latere versie gepubliceerd door de Free Software Foundation; zonder invariante paragrafen, zonder vooromslagteksten, en zonder achteromslagteksten.

*[Over deze site](#) | Laatst gewijzigd: 23 juni 2004, 21:29 door: Tom Verbreyt*